Report5

Load the R packages

```
library(knitr)
library(readr)
library(tidyverse)
library(dplyr)
library(tidyr)
library(stringr)
library(purrr)
library(httr)
library(tibble)
```

Part 1

Sending some example queries to Movie DB with my API key:

```
# What are the highest grossing dramas from 2010?
rm(data, data_content)
data <-  httr::GET("https://api.themoviedb.org/3/discover/movie?%with_genres=18&primary_release_year=20
data_content <- httr::content(data)
```

The 3 highest grossing dramas from 2010 are The Twilight Saga: Eclipse, The King's Speech, The Karate Kid.

```
# Have Will Ferrell and Liam Neeson even been in a movie together?
rm(data, data_content)
data <-  httr::GET("https://api.themoviedb.org/3/discover/movie?&with_cast=3896&with_cast=23659&api_key=
data_content <- httr::content(data)
```

After inspecting the content of the GET response, the two actors Will Ferrell and Liam Neeson have not been in a movie together.

```
# Can you find kids movies with Tom Cruise in it?
rm(data, data_content)
data <-  httr::GET("https://api.themoviedb.org/3/discover/movie?&with_cast=500&certification_country=US
data_content <- httr::content(data)
```

After inspecting the content of the GET response, the Tom Cruise has not featured in a kids movie.

Part 2

Search for the ID of "Pixar":

```
rm(data, data_content)
data <-  httr::GET("https://api.themoviedb.org/3/search/company?api_key=15079d94fcca66e5a038ec57e3f3200
data_content <- httr::content(data, as = "parsed")
```

The Company ID is 3

Part 3

Here we write a query that will search all the Pixar movies, which are then put into a tibble:

```r
# Get the Pixar movies and sort them by descending revenue
rm(data, data_content)
data <-  httr::GET("https://api.themoviedb.org/3/discover/movie?%with_companies=3&sort_by=revenue.desc&a
data_content <- httr::content(data, as = "parsed")
# Convert them to a tibble
n_movies <- 20
titles <- vector("character", length(n_movies))
popularity <- vector("character", length(n_movies))
for (i in 1:n_movies) {
  titles[i] <- purrr::pluck(data_content, "results", i, "original_title")
  popularity[i] <- purrr::pluck(data_content, "results", i, "popularity")
}
movie_tibble <- as_tibble(titles) %>%
  rename(TITLE = value) %>%
  mutate(POPULARITY = popularity)
movie_tibble
```

```
## # A tibble: 20 x 2
##    TITLE               POPULARITY
##    <chr>               <chr>
##  1 Incredibles 2       36.67
##  2 Toy Story 3         25.606
##  3 Finding Dory        17.582
##  4 Finding Nemo        28.81
##  5 Inside Out          36.258
##  6 Coco                35.225
##  7 Monsters University 19.141
##  8 Up                  20.879
##  9 The Incredibles     28.865
## 10 Ratatouille         25.89
## 11 Monsters, Inc.      28.712
## 12 Cars 2              23.492
## 13 Brave               30.677
## 14 WALL·E              26.831
## 15 Toy Story 4         56.105
## 16 Toy Story 2         30.033
## 17 Cars                132.003
## 18 Cars 3              20.832
## 19 Toy Story           37.616
## 20 A Bug's Life        27.824
```

The extraction of the movie title and the popularity could have been done with a combination of the map() and pluck() command, but for I opted for the sort of archaic solution with a simple for-loop.